[ABSTRACT]

Disclosed is a high speed motion estimation method for real time image compression in an apparatus for compressing image data.

According to the present invention, a typical three-step search method using UESA is improved to use the less number of search points and the improved three step search method is properly applied to a half-stop method having a matching error threshold and a partial error sum. Therefore, motion can be estimated at high speed using the high speed motion estimation method according to the present invention. Also, it is possible to perform calculations in high speed by reducing search points while estimated image quality is sustained identically to that of a typical three step search method in order to be applied into a software based real time system.

[REPRESENTATIVE DRAWING]

FIG. 10

[INDEX WORD]

MPEG, Motion estimation, motion estimator, motion vector, three step search

[SPECIFICATION]

[TITLE OF THE INVENTION]

HIGH SPEED MOTION ESTIMATION METHOD FOR REAL TIME IMAGE COMPRESSION

[BRIEF DESCRIPTION OF THE DRAWINGS]

FIG. 1 is a diagram illustrating a typical image data transceiving system;

FIG. 2 is a block diagram illustrating a MPEG source encoder included in the transmitter shown in FIG. 1;

FIG. 3 is a diagram showing a blocked frame image;

FIG. 4 shows a current frame image that is a partially blocked frame image of FIG. 3;

FIG. 5 is a diagram illustrating a previous frame image having a block corresponding to the current frame image of FIG. 4;

FIG. 6a shows a location of a current block, which is initially searched at a search window, and FIG. 6b shows a search direction of a current block in a search window;

FIG. 7 is a diagram illustrating a search window size;

FIGS. 8a to 8i show search points in a search window, which are decided by a conventional three step search method;

FIGS. 9a to 9c are search point diagrams for describing a conventional three step search method;

FIG. 10 is a flowchart of a high speed motion estimation method according to an

embodiment of the present invention;

FIG. 11 is a flowchart illustrating a step for searching search points in a first step search of FIG. 10;

FIG. 12 is a flowchart illustrating a second step search with a search point having the smallest SAD of FIG. 11 as a center;

FIG. 13 is a flowchart illustrating a step for searching search points in a second step search of FIG. 12;

FIG. 14 is a flowchart illustrating a third step search with a search point having the smallest SAD of FIG. 13 as a center;

FIGS. 15a to 15f are search point diagrams for the flowchart of FIG. 10 and FIG. 11;

FIG. 16 is a search point diagram for the flowchart of FIG. 12 and FIG. 13; and

FIGS. 17a to 17b are search point diagrams for the flowchart of FIG. 14.

<DESCRIPTION OF THE SYMBOLS IN MAIN PORTIONS OF THE DRAWINGS>

121: Motion estimator
F(t-1): previous frame image
SRW: search window
SAD: sum of absolute differences
SAD(x): SAD of a search point x
f(t): current frame image
BC: current block
MV: motion vector
SADmin: smallest value of SAD
TH: matching error threshold

[DETAILED DESCRIPTION OF THE PRESENT INVENTION]

[OBJECT OF THE PRESENT INVENTION]

[FIELD OF THE INVENTION AND DESCRIPTION OF THE RELATED ART]

The present invention relates to a high speed motion estimation method for real time image compression in an apparatus for compressing image data. More particularly, the present invention relates to a high speed motion estimation method for real time image compression by improving a three-step search method using UESA to reduce the number of search points and properly applying the improved three-step search method to a half-stop method having a threshold for a matching error and a partial error sum.

Since the size of image data is generally large compared to voice and text data, it is impossible to process the image data in real time unless the image data is compressed. That is, the compression of the image data makes it possible to process image signal in real time, for transmitting or storing the image data in real time. As the international standards for compressing images, JPEG was introduced for compressing a still image, MPEG1 was introduced for television broadcasting, MPEG2 was introduced for satellite broadcasting, and MPEG4 has been developed for low speed bit rate transmission.

The image data is compressed by removing redundant data. The data redundancy is caused because image data itself is different from data for expressing image information.

As described above, the data redundancy includes spatial redundancy in one frame image, probabilistic redundancy, and temporal redundancy of frame images. The similarity of adjacent pixels causes the spatial redundancy. That is, the spatial redundancy means that a pixel has a value similar to that of adjacent pixel. Discrete Cosine Transform (DCT) is used to process the spatial redundancy. The DCT gathers image information at an upper left side of a frame image.

The similarity of symbols causes the probabilistic redundancy. That is, the probabilistic redundancy means that data is not uniformly distributed and a symbol has a value similar to that of adjacent symbol. Variable length coding, entropy coding

scheme, is used to process the probabilistic redundancy. The variable length coding allocates a bit size corresponding to the size of a symbol.

The similarity between a current frame image and a previous frame image causes temporal redundancy. Motion estimation/motion compensation (ME/MC) is used to process the temporal redundancy. The motion estimation (ME) detects motion vectors of a current frame image and a previous frame image, generates new frame image by compensating the motion using the detected motion vector, and removes identical data between the current frame image and the previous frame image by subtracting the new frame image from the current frame image.

FIG. 1 is a diagram illustrating a typical image data transceiving system. Referring to FIG. 1, the image data transceiving system includes an image data transmitter 100 for compressing image data and transmitting the compressed image data, a satellite 1 for receiving an image signal from the image data transmitter 100 and transmitting the received image signal to a receiver 1, and an image data receiver 200 for receiving and uncompressing the image signal from the satellite 1, thereby restoring the uncompressed image signal to original image data.

The image data transmitter 100 includes a MPEG source encoder 110 for compressing image data and sound data, a text encoder 130 for compressing text, a channel encoder 150 for including channel information into the encoded data to overcome noise problem, a radio frequency (RF) unit 170 for modulating the encoded data. The image data receiver 200 includes a baseband processor 210 for restoring the image signal from the satellite 1 to base band image data through removing carrier wave, a channel decoder 220 for detecting errors, correcting and restoring the image data from the baseband processor 210, and a MPEG decoder 230 for decompressing the image data from the channel decoder 220 to restore the original data.

FIG. 2 is a block diagram illustrating a MPEG source encoder included in the transmitter shown in FIG. 1.

Referring to FIG. 2, the MPEG source encoder includes a 8x8 blocking unit 111

for dividing data of one frame image into 8x8 blocks, a subtractor 112 for subtracting a new frame image from the current frame image from the 8x8 blocking unit 111, a 8x8 DCT unit 113 for performing discrete cosine transform (DCT) on the current frame image from the subtracter 112, a 8x8 quantizing unit for quantizing the frame image from the 8x8 DCT unit 113, a variable length coding unit 115 for performing variable length coding on the current frame image from the 8x8 quantizing unit 114, a 8x8 inverse-quantizing unit for performing inverse-quantizing on the frame image from the 8x8 quantizing unit 114, a 8x8 inverse-DCT unit 118 for performing inverse-DCT on the frame image from the 8x8 inverse-quantizing unit 117, an adder 119 for adding the frame image from the 8x8 inverse-DCT unit 118 and a newly generated frame image, a memory 120 for storing frame images from the adder 119, a 16x16 blocking unit 123 for dividing the input frame image into 16x16 blocks, a motion estimation unit 1212 for estimating a motion vector by comparing pixel values of the current frame image from the 16x16 blocking unit 123 with that of a previous frame image from the frame memory 120, a motion compensator 122 for generating a new frame image by applying the estimated motion vector from the motion estimator 121 to the frame image of the frame memory 120, and a multiplexer 116 for multiplexing the image data from the 8x8 variable length coding unit 115 and the motion vector of the motion estimator 121.

Meanwhile, the resolution of one frame image (horizontal pixel number x vertical pixel number may be diverse, for example, 720 x 480, and 1192 x 1080. The motion estimator for estimating a motion vector between a current frame image and a previous frame image divides one frame image into a block having 16x16 pixels and processes the one frame image by block.

The motion estimator 121 estimates the variation of motions by comparing the pixel values of the current frame image (f(t)) with the pixel values of the previous frame image (f(t-1)). Such an operation of the motion estimator 121 will be described in more detail.

FIG. 3 is a diagram showing a blocked frame image. The 16x16 blocking unit

123 of FIG. 2 divides one frame image by a block having 16x16 pixels as a unit (hereinafter a 16x16 block). That is, the blocked frame image is shown in FIG. 3.

FIG. 4 shows a current frame image that is a partially blocked frame image of FIG. 3. That is, FIG. 4 shows a current frame image (f(t)) having eight blocks around a current block (B(t)22) as a center, total 9 blocks. FIG. 5 is a diagram illustrating a previous frame image having a block corresponding to the current frame image of FIG. 4. That is, FIG. 5 shows the previous frame image (f(t-1)) having total 9 blocks, 8 blocks around a block (B(t-1)22) corresponding to the current block (B(t)22) of FIG. 4. In FIG. 5, a dotted line part is a search window (SRW) having a plurality of blocks identical to the current block (B(t)22). The SRW is decided in consideration of a movable range of a motion between two consecutive frame images in about 24 frame images per a second. In general, an extension range (± block size/2) is applied to the corresponding block (B(t-1))22).

Referring to FIG. 5, the motion estimator 121 of FIG. 2 compares the current block (B(t) 22) of FIG. 4 with a plurality of blocks in the SRW of FIG. 5. A comparison direction is from an upper left end to a lower right end as shown in FIG. 6A, and it is identical to a scan direction of an electron gun of a cathode-ray tube as shown in FIG. 6B. As described above, the motion vector is estimated by finding a matching block most similar to the current block. That is, an algorithm for finding a matching block most similar to the current block is referred as a block matching algorithm.

The block matching algorithm compares pixel values of blocks. That is, the pixel value of a current block of a current frame image is compared with that of a corresponding block of a previous frame image. To find a matching block is to estimate a motion vector by finding a block having the smallest difference through subtracting the pixel value of a corresponding block from the pixel value of a current block and calculating a location vector of a matching block with a current block as a center based on the found block.

Meanwhile, the motion vector estimation is performed in consideration of the

prevention of image quality degradation and high speed estimation.

The block matching algorithm includes various methods. Among them, a full search method is used as an evaluation reference of other methods. However, the full search method needs total 24bits, 8 bits per each three primary colors (R, G, B), in order to express one color pixel although the estimated image quality is very good. That is, the full search method has the large computation amount to process all pixel values. Therefore, it is impossible to embody a real time system with the full search method.

In order to overcome such a drawback of the full search method, many high speed matching algorithms have been introduced. Such high speed matching algorithms includes a method for reducing searching points by unimode error surface assumption (UESA), a multi-resolution method, a method for moving a reference point using correlation between adjacent motion vectors and variable search range (VSR), and a method for detecting calculations in a block matching operation.

A three-step search method is the representative method of the method for reducing searching points by UESA. Hereinafter, the three-step search method will be described in detail.

FIG. 7 is a diagram illustrating a search window size. Referring to FIG. 7, the search window (SRW) is larger than a reference block by applying an expansion range in top and bottom and left and right direction to the reference block having 16 x 16 pixels. In the expansion range, a size of expanding a reference block in one direction is + block size/2. Hereinafter, +7 is applied as the size of expanding a reference block in one direction as shown in FIG. 7.

FIG. 8a to 8i is a diagram illustrating a searching point in a search window, which is decided by a conventional three-step search method. The conventional three-step search method searches a matching block by comparing with nine blocks as shown in FIG. 8. A center point of nine blocks to be searched in a search window is expressed as search points 1 to 9, as shown in FIG. 8.

FIG. 9A to FIG. 9C are search point diagrams for describing a conventional three step search method (TSS). In case of searching an entire area in a search window, search points that can be a center point of a block are displayed at first, and 9 search pints decided by the three-step search method are displayed as shown in FIG. 9A to FIG. 9C.

With reference to FIG. 9A to FIG. 9C, the three-step search method will be described.

At first, a sum of absolute differences (SAD) is used to find a target matching block in consideration of the computation complexity and the performance. The SAD is the sum of differences between pixel values of a current block of a current frame image (f(t)) and pixel values of a block corresponding to the current block in a search wind of a previous frame image (f(t-1)). The SAD can be calculated like Equation 1.

$$SAD(x) = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} | Ic(k+i, 1+j) - Ip(k+x+i, 1+y+j) \qquad \text{Equation 1}$$

In Equation 1, Ic() denotes a pixel value of a block of a current image block, and Ip() is a pixel value for a block corresponding to a search window of a previous frame image. x, y is a coordinate of a search window, k, 1 is a coordinate of a corresponding block, and n is a size of a matching block.

A motion vector is decided through first, second, and third searching steps using unimode error surface assumption (UESA) that the difference between search points increases as the search point becomes further far from a global motion vector. At the first search step, SADs for all of 9 search points are calculated as shown in FIG. 9A. At the second search step, the smallest SAD is selected from the calculated SADs from the first search step. If the smallest SAD is 2, SADs for 9 search points from 21 to 29 are calculated using the search point 2 as a center. At the third search step, the smallest SAD is selected from the SADs calculated at the second search step. If the smallest SAD is 22, SADs for 9 search points are calculated as the search point 22 as a center.

Then, the smallest SAD is selected from the newly calculated SADs, and the smallest SAD is decided as a motion vector.

Such a conventional three-step search method performs search operations for nine search point modes for each of the first, second, and third steps. In this method, total 25 search points are searched, such as the nine search points at the first step, eight search points at the second step, and eight search points at the third step. In consideration of 24bits required to express one pixel value, the search time increase due to too many search points to calculate. Therefore, it is impossible to process in a software manner.

[TECHNICAL OBJECT OF THE INVENTION]

It is therefore an object of the present invention to provide a high speed motion estimation method for real time image compression by improving a three-step search method using UESA to reduce the number of search points and properly applying the improved three-step search method to a half-stop method having a threshold for a matching error and a partial error sum.

It is another object of the present invention to a high speed motion estimation method for real time image compression, which can perform calculations in high speed by reducing search points while estimated image quality is sustained identically to estimated image quality of a typical three step search method in order to be applied into a software based real time system.

[CONSTITUTION AND OPERATION OF THE INVENTION]

To achieve the above objects and other advantages, a high speed motion estimation method for reducing search points in a three step search method using unimode error surface assumption (UESA) includes the steps of: calculating a SAD of a center search point and SADs of two adjacent search points among four search points disposed at top, bottom, left, and right of the center search point, finding a

corresponding search point having the smallest SAD among the three calculated SADs, calculating SADs of another two search points adjacent to a search point having the smallest SAD if the search point having the smallest SAD is one of the two adjacent search points, and performing a second step search for a search point having the smallest SAD among the three SADs.

If a search point having the smallest SAD is a center search point, SADs of another two adjacent search points among four search points disposed at the top, bottom, left, and right of the center search point, finding a search point having the smallest SAD among the calculated SADs of the three search points. If the search point having the smallest SAD is one of the two adjacent search points, SADs of another search points adjacent in both directions are calculated, and a second step search is performed for a search point having the smallest SAD among the calculated SADs of the three search points.

On the contrary, if a center search point has the smallest SAD, the second step search is performed with the center search point as a center. Rest steps of the second step search are identical to the first step search, and the third step search according to the present invention is identical to that of the conventional three step search.

In the first, the second, and the third step search, a calculated SAD is compared to a preset SADmin while SADs of search points are calculating. If the calculated SAD is larger than the SADmin, the SAD calculation process stops and the next step is performed. After one SAD of a search point is completely calculated, if the SAD is smaller than a matching error threshold (TH), the corresponding search point of the SAD is decided as a motion vector. Also, if the calculated SAD is smaller than the initially set SADmin, the SADmin is reset to the calculated SAD, thereby increasing the probability of interrupting the SAD calculation. Therefore, the motion estimation can be performed in high speed by reducing the amount of computation.

The present invention now will be described more fully hereinafter with reference

to the accompanying drawings, in which preferred embodiments of the invention are shown. This invention may, however, be embodied in many different forms and should not be construed as limited to the embodiments set forth herein; rather, these embodiments are provided so that this disclosure will be thorough and complete, and will fully convey the scope of the invention to those having skill in the art.

A high speed motion estimation method for estimating a motion vector by detecting a matching block that matches with a current block of a current frame image (f(t)) from a plurality of blocks in a search window of a previous frame image (f(t-1)) is a method modified from a typical three-step search method for reducing search points in a motion estimator of an encoder for compressing image data.

The first step of the three-step search method will be described with reference to FIG. 15. At first, search points 5, 6, and 8 are searched as shown in FIG. 15B. If the SAD of the search point 6 is the smallest value, the SADs of search points 3, 6, and 9 again. Then, the second step is performed with the search point having the smallest SAD as a center point. If the SAD of the search point 8 is the smallest value as shown in FIG. 15C, the SADs of search points 7, 8, and 9 are calculated again. Then, the second step is performed with the search point having the smallest SAD as a center point. On the contrary, if the search point 5 has the smallest SAD value, SADs of the search points 2 and 4 are calculated again as shown in FIG. 15D. If the search point 4 has the smallest SAD value among the CADs of the search points 2, 4, and 5, SADs of search points 1, 4, and 7 are calculated again as shown in FIG. 15E. Then, the second step is performed with the search point having the smallest SAD. If the search point 2 has the smallest SAD among the search points 2, 4, and 5, SADs of search points 1, 2, and 3 are calculated again as shown in FIG. 15F. Then, the second step is performed with the search point having the smallest SAD as a center point. If the search point 5 has the smallest SAD value among the search points 2, 4 and 5, the second step is performed with the search point 5 as a center point.

The three step method of the high speed motion estimation method according to

the present embodiment will be described with reference FIG. 10, FIG. 11, and FIG. 15.

The step (a) of the first step search in the three-step search method according to the present embodiment will be described hereinafter. Referring to the steps 51 and 52 in FIG. 10, a matching error threshold (TH) is set, and the SAD of a search point in a search window is calculated. Then, the calculated SAD is set as SADmin. If the SADmin is smaller than the set matching error threshold (TH), the search point of the SADmin is decided as a motion vector. If not, the SAD calculation step is continuously performed.

The predetermined SADmin is the SAD of a predetermined search point in a SRW of a previous frame image (f(t-1)). The predetermined SADmin is set in consideration of a center search point of a corresponding block or a previous motion vector.

Referring to the step 53 in FIG. 10, the SAD of a center search point 5 is calculated among nine search points 1 to 9 in the first step SRW of a previous frame image (f(t-1)) as shown in FIG. 15, and the SADs of two adjacent search points 6 and 8 are calculated among four search points 2, 4, 6, and 8 disposed at upper, lower, left, and right of the center search point 5 as shown in FIG. 15A.

While the SADs of the search points 5, 6, and 8 are calculating, it is determined whether each of the calculated SADs is larger than the pre-set SADmin. If the calculated SAD is not larger than the pre-set SADmin, the SAD calculation is continuously performed. If the calculated SAD is larger than the pre-set SADmin, the SAD calculation stops, and the next step is performed.

Referring to the step 54 in FIG. 10, it is determined whether the calculated SADs include a SAD smaller than the matching error threshold (TH) or not. If the calculated SADs include a SAD smaller than the matching error threshold, the search point of the SAD smaller than the matching error threshold is decided as a motion vector. If not, the smallest SAD is selected from the calculated SADs at step 55 of FIG. 10.

The step (b) in the first step search of the three-step search method will be

described. Referring to the step S55 of FIG. 10, the smallest SAD found in the step (a) is one of the search points 6 and 8. If the smallest SAD is the search point 6, it is determined whether the SAD of the search point 6 is smaller than SADmin or not. If the SAD of the search point 6 is smaller than SADmin, the SAD of the search point 6 is set as the SADmin at steps 61 and 62 of FIG. 11.

Then, SADs of two search points adjacent to the search point having the smallest SAD are calculated. For example, if the search point 6 has the smallest SAD, the SADs of the search points 3 and 9 are additionally calculated as shown in the step 63 of FIG. 11 and FIG. 15B. While the SADs of the search points 3 and 9 are calculating, the calculated SAD is compared with the predetermined SADmin. If the calculated SAD is larger than the predetermined SADmin, the SAD calculation process stops and the next step is performed. IF not, the SAD calculation process is continuously performed.

On the contrary, if the search point 8 has the smallest SAD, it is determined whether the SAD of the search point 8 is compared with the predetermined SADmin. If the SAD of the search point 8 is smaller than the predetermined SADmin, the SAD of the search point 8 is set as the SADmin. Then, as shown in FIG. 15C, the SADs of the search points 7 and 9 are additionally calculated. While the SADs of the search points 7 and 9 are calculating, the calculated SAD is compared with the predetermined SADmin. If the calculated SAD is larger than the predetermined SADmin, the SAD calculation process stops and the next step is performed. If not, the SAD calculation process is continuously performed.

If the SADs of the search points 6 and 8 are not the smallest SAD, and if the search point 6 has the smallest SAD, it is determined whether the SAD of the search point 5 is smaller than the predetermined SADmin or not at step 56 of FIG. 10. If the SAD of the search point 5 is smaller than the predetermined SADmin, the SAD of the search point 5 is set as the SADmin. Then, a SAD of a center search point 5 is calculated among nine search points 1 to 9 in the first step search window of a previous

frame image (f(t-1)), and SADs of two other adjacent search points 2 and 4 are calculated among four search points 2, 4, 6, and 8 disposed at the top, bottom, left, and right of the center search point 5 at the step 57 of FIG. 10.

While the SADs of the search points 5, 2 and 4 are calculated, it is determined whether the calculated SAD of one of the search points 5, 2, and 4 is larger than the predetermined SADmin or not. If the calculated SAD is not larger than the predetermined SADmin, the SAD calculation process is continuously performed. If not, the SAD calculation process stops, and the next step is performed.

Then, it is determined whether the calculated SADs include a SAD smaller than the matching error threshold (TH) or not. If the calculated SADs include the SAD smaller than the matching error threshold, the corresponding search point having the SAD smaller than the matching error threshold is decided as a motion vector. If not, the smallest SAD is searched among the calculated SADs at step 59 of FIG. 10.

Referring to the step 64 of FIG. 11, it is determined whether the calculated SAD is smaller than the matching error threshold (TH) or not. If the calculated SAD is smaller than the matching error threshold, the search point of the corresponding SAD is decided as a motion vector. Referring to the step 65 of FIG. 11, one having the smallest SAD value is selected from the search points 3, 6, and 9 (or the search points 7, 8. and 9). Referring to the step 65 of FIG. 11, the second search step for the second step search window is performed with the selected search point having the smallest SAD value as a center search point.

As described above, the smallest SAD value obtained in the step (b) is the SAD value of one of the search points 3, 6, and 9 or the SAD value of one of the search points 7, 8, and 9. And, the second search step will be described when the search point 3 has the smallest SAD value with reference to FIG. 12, FIG. 13, and FIG. 16. Since the search method for corresponding search points is identical, the detailed description of the second search step for other search points is omitted.

In the second search step with the search point 3 as a center, eight search points

are search points 31, 32, 33, 34, 36, 37, 38 and 39.

Referring to the steps 71 and 72 of FIG. 12, if the search point 3 has the smallest SAD value, it is determined whether the SAD value of the search point 3 is smaller than a predetermined SADmin or not. If the SAD value of the search point 3 is smaller than the predetermined SADmin, the SAD value of the search point 3 is set as the SADmin.

Referring to the step 73 of FIG. 12, a SAD value of a center search point 3 is calculated among nine search points 31 to 39 in the second search window having the search point 3 as a center as shown in FIG. 16(a). And, SAD values of two adjacent search points 36 and 38 are calculated among the four search points 32, 34, 36, and 38 disposed at the top, bottom, left, and right of the center search point 3. Then, the smallest SAD is selected from the calculated SADs of the search points 3, 36, and 38 (SAD(3), SAD(36) and SAD(38)).

In the three-step search method according to the present embodiment, the second search step is identical to the first search step only but the sizes of the search windows are different like the first step search window shown in FIG. 15 and the second step search window shown in FIG. 16.

While the SADs are calculating, it is determined whether a calculated SAD is larger than the SADmin or not. If the calculated SAD is larger than the predetermined SADmin, the SAD calculation process stops and the next step is performed.

It is determined whether the calculated SAD is smaller than the matching error threshold (TH). If the calculated SAD is smaller than the matching error threshold, the corresponding search point having the SAD smaller than the matching error threshold is decided as a motion vector.

As shown in FIG. 13, the calculated smallest SAD is one of SAD(36) and SAD(38). If the calculated smallest SAD is SAD(36), SADs of two search points 33 and 39 adjacent to the corresponding search point 36 are calculated. If the calculated smallest SAD is SAD(38), SADs of two search points 37 and 39 adjacent to the corresponding search point 38 are calculated. As described above, the smallest SAD

is selected from the calculated SADs of three search points, and the third step search is performed for the third step search window with the corresponding search point having the selected smallest SAD.

On the contrary, if the search point 3 has the smallest SAD value, SAD values of other two adjacent search points 32 and 34 are calculated among the four search points 32, 34, 36, and 38 disposed at the top, bottom, left, and right of the search point 3. Then, the smallest SAD is selected from the calculated SADS of the search points 32 and 34 and the SAD of the center search point 3.

If the smallest SAD is one of SAD(32) and SAD(34), for example, if the smallest SAD is SAD(32), SADS of two search points 31 and 33 adjacent to the search point 32 are calculated. On the contrary, if the smallest SAD is SAD(34), SADS of two search points 31 and 37 adjacent to the search point 34 are calculated. As described above, the smallest SAD is selected from the calculated SADs of three search points. Then, the third step search is performed for the third step search window using the selected smallest SAD as a center search point.

On the contrary, if the search point 3 has the smallest SAD, the third step search is performed for the third step search window using the search point 3 as the center search point.

Hereinafter, the third step search with a search point 36 having the smallest SAD will be described with reference to FIG. 14, and FIG. 17. Since the third step search method is identical to other search point, the detailed description thereof is omitted.

If the search point 38 has the smallest SAD, SADs of all search points in the third step search window are calculated. That is, the SADs of nine search points including the search point 38 are calculated. Then, a search point having the smallest SAD is selected from the calculated SADs and the selected search point is decided as a motion vector.

It will be apparent to those skilled in the art that various modifications and variations can be made in the present invention. Thus, it is intended that the present

invention covers the modifications and variations of this invention provided they come within the scope of the appended claims and their equivalents.

[EFFECT OF THE INVENTION]

As described previously, in the high speed motion estimation method according to the present invention, a typical three-step search method using UESA is improved to use less number of search points and the improved three step search method is properly applied to a half-stop method having a matching error threshold and a partial error sum. Therefore, motion can be estimated at high speed using the high speed motion estimation method according to the present invention.

The high speed motion estimation method according to the present invention can perform calculation operations in high speed by reducing search points while estimated image quality is sustained identically to that of a typical three step search method. Therefore, the high speed motion estimation method according to the present invention can be applied to a software based real time system.

This invention has been described above with reference to the aforementioned embodiments. It is evident, however, that many alternative modifications and variations will be apparent to those having skill in the art in light of the foregoing description. Accordingly, the present invention embraces all such alternative modifications and variations as fall within the spirit and scope of the appended claims.

WHAT IS CLAIMED IS:

1. A high speed motion estimation method for estimating a motion vector by detecting a block matched with a current block of a current frame image (f(t)) among a plurality of blocks in a search window of a previous frame image (f(t-1)), which is performed in a motion estimator of an encoder compressing image data, the high speed motion estimation method comprising the steps of:

a) calculating a SAD (sum of absolute differences) of a center search point 5 among nine search points 1 to 9 in a first step search window of a previous frame image (f(t-1)) and SADs of two adjacent search points 6 and 8 among four search points 2, 4, 6, and 8 disposed at the top, bottom, left, and right of the search point 5 using a three-step search method, and selecting the smallest SAD from the calculated SADs;

b) calculating two search points adjacent to the search point having the selected smallest SAD at the step a) if the selected smallest SAD is one of SADs of two search points 6 and 8, finding the smallest SAD from the SADs of the three search points, and performing a second step search for a second step search window using a corresponding search point having the found smallest SAD;

c) calculating SADs of two other adjacent search points 2 and 4 among search points 2, 4, 6, and 8 disposed at top, bottom, left, and right of a search point 5 if the found smallest SAD is a SAD of a center search point 5 at step a), and finding a smallest SAD from the SADs of two search points 2 and 4, and the SAD of the center search point 5;

d) calculating SADs of two search points adjacent to a search point having the smallest SAD if the found smallest SAD is one of the SADs of two search points 2 and 4, finding a smallest SAD from the SADs of three search points, and performing a second step search for a second search window using a corresponding search point having the found smallest SAD; and

e) performing a second step search for a second search window with a center

search point 5 if the center search point 5 has the found smallest SAD at step c).

2. The high speed motion estimation method of claim 1, further comprising the step of f) setting a SADmin by calculating a SAD for a preset search point in a search window (SRW) of a previous frame image (f(t-1)).

3. The high speed motion estimation method of anyone of claims 1 and 2, further comprising the steps of: g) determining whether a SAD(5) is smaller than the SADmin set at the step f) or not if the smallest SAD obtained at the step e) is the SAD of the search point 5, and resetting the SADmin with the SAD(5).

4. The high speed motion estimation method of claim 1, further comprising the step of h) setting a matching error threshold (TH).

5. The high speed motion estimation method of claim 2, wherein in the step a) it is determined whether a calculated SAD is larger than the SADmin while SADs of search points are calculating, the SAD calculation process stops if the calculated SAD is larger than the SADmin, and a next step is performed.

6. The high speed motion estimation method of claim 2, wherein in the step b) it is determined whether a calculated SAD is larger than the SADmin while SADs of search points are calculating, the SAD calculation process for the corresponding search point stops if the calculated SAD is larger than the SADmin, and a next step is performed.

7. The high speed motion estimation method of claim 2, wherein in the step c) it is determined whether a calculated SAD is larger than the SADmin while SADs of search points are calculating, the SAD calculation process for the corresponding search

point stops if the calculated SAD is larger than the SADmin, and a next step is performed.

8. The high speed motion estimation method of claim 2, wherein in the step d), it is determined whether a calculated SAD is larger than the SADmin while SADs of search points are calculating, the SAD calculation process for the corresponding search point stops if the calculated SAD is larger than the SADmin, and a next step is performed.

9. The high speed motion estimation method of claim 4, wherein in the step a), it is determined whether the calculated SAD is smaller than the predetermined matching error threshold (TH) or not, and a corresponding search point of the calculated SAD is set as a motion vector if the calculated SAD is smaller than the matching error threshold.

10. The high speed motion estimation method of claim 4, wherein in the step b), it is determined whether the calculated SAD is smaller than the predetermined matching error threshold (TH) or not, and a corresponding search point of the calculated SAD is set as a motion vector if the calculated SAD is smaller than the matching error threshold.

11. The high speed motion estimation method of claim 4, wherein in the step c), it is determined whether the calculated SAD is smaller than the predetermined matching error threshold (TH) or not, and a corresponding search point of the calculated SAD is set as a motion vector if the calculated SAD is smaller than the matching error threshold.

12. The high speed motion estimation method of claim 4, wherein in the step

d), it is determined whether the calculated SAD is smaller than the predetermined matching error threshold (TH) or not, and a corresponding search point of the calculated SAD is set as a motion vector if the calculated SAD is smaller than the matching error threshold.

13. The high speed motion estimation method of claim 4, wherein in the step e), it is determined whether the calculated SAD is smaller than the predetermined matching error threshold (TH) or not, and a corresponding search point of the calculated SAD is set as a motion vector if the calculated SAD is smaller than the matching error threshold.

14. The high speed motion estimation method of claim 1, wherein the step b) includes the steps of:

b1) calculating SADs of search points 3 and 9 if a search point 6 has the smallest SAD, and finding a smallest SAD among SAD(3), SAD(9), and SAD(6);

b2) performing a second step search for a second step search window using a search point 3 as a center search point if the smallest SAD is SAD(3);

b3) performing a second step search for a second step search window using a search point 6 as a center search point if the smallest SAD is SAD(6); and

b4) performing a second step search for a second step search window using a search point 9 as a center search point if the smallest SAD is SAD(9).

15. The high speed motion estimation method of claim 1, wherein the step b) includes the steps of:

b1) calculating SADs of search points 7 and 9 if a search point 8 has the smallest SAD, and finding a smallest SAD among SAD(7), SAD(9), and SAD(8);

b2) performing a second step search for a second step search window using a search point 7 as a center search point if the smallest SAD is SAD(7);

b3) performing a second step search for a second step search window using a search point 8 as a center search point if the smallest SAD is SAD(8); and

b4) performing a second step search for a second step search window using a search point 9 as a center search point if the smallest SAD is SAD(9).

16. The high speed motion estimation method of anyone of claims 14 and 15, wherein the step b) further includes the step of:

b5) determining whether a predetermined SADmin is smaller than SAD(6) or not if a search point 6 has the smallest SAD, and resetting a predetermined SADmin to SAD(6) if the predetermined SADmin is smaller than SAD(6), and

determining whether a predetermined SADmin is smaller than SAD(8) or not if a search point 8 has the smallest SAD, and resetting a predetermined SADmin to SAD(8) if the predetermined SADmin is smaller than SAD(8).

17. The high speed motion estimation method of anyone of claims 14 and 15, wherein in the step b1, it is determined whether a calculated SAD is larger than a SADmin or not while SADs of search points are calculating, the SAD calculation process stops if the calculated SAD is larger than the SADmin, and a next step is performed.

18. The high speed motion estimation method of anyone of claims 14 and 15, wherein in the step b1), it is determined whether a calculated SAD is smaller than a predetermined matching error threshold or not, and a corresponding search point of the SAD smaller than the matching error threshold is decided as a motion vector.

19. The high speed motion estimation method of claim 1, wherein the step d) includes the steps of:

d1) calculating SADs of search points 1 and 3 if a search point 2 has the smallest SAD, and finding a smallest SAD among SAD(1), SAD(3), and SAD(2);

d2) performing a second step search for a second step search window using a search point 1 as a center search point if the smallest SAD is SAD(1);

b3) performing a second step search for a second step search window using a search point 2 as a center search point if the smallest SAD is SAD(2); and

b4) performing a second step search for a second step search window using a search point 3 as a center search point if the smallest SAD is SAD(3).

20. The high speed motion estimation method of claim 1, wherein the step d) includes the steps of:

d1) calculating SADs of search points 1 and 7 if a search point 4 has the smallest SAD, and finding a smallest SAD among SAD(1), SAD(7), and SAD(4);

d2) performing a second step search for a second step search window using a search point 1 as a center search point if the smallest SAD is SAD(1);

b3) performing a second step search for a second step search window using a search point 7 as a center search point if the smallest SAD is SAD(7); and

b4) performing a second step search for a second step search window using a search point 4 as a center search point if the smallest SAD is SAD(4).

21. The high speed estimation method of anyone of claims 19 and 20, wherein the step d) further includes the step of:

d5) determining whether a predetermined SADmin is smaller than SAD(2) or not if a search point 2 has the smallest SAD, and resetting a predetermined SADmin to SAD(2) if the predetermined SADmin is smaller than SAD(2), and

determining whether a predetermined SADmin is smaller than SAD(4) or not if a search point 4 has the smallest SAD, and resetting a predetermined SADmin to SAD(4) if the predetermined SADmin is smaller than SAD(4).

22. The high speed motion estimation method of anyone of claims 19 and 20,

wherein in the step d1), it is determined whether a calculated SAD is larger than a SADmin or not while SADs of search points are calculating, the SAD calculation process stops if the calculated SAD is larger than the SADmin, and a next step is performed.

23. The high speed motion estimation method of anyone of claims 19 and 20, wherein in the step d1), it is determined whether a calculated SAD is smaller than a predetermined matching error threshold or not, and a corresponding search point of the SAD smaller than the matching error threshold is decided as a motion vector.

24. The high speed motion estimation method of claim 1, wherein the second step search for the second step search window in the step e) includes the steps of:

e1) calculating a SAD of a center search point 5 among nine search points 51 to 59 in the second step search window having the search point 5 as a center, calculating SADs of two adjacent search points 56 and 57 among search points 52, 54, 56, and 58 disposed at a top, bottom, left, and right of the center search point 5, using a three-step search method and finding a smallest SAD among SAD(5), SAD(56), and SAD(58);

e2) calculating SADs of two search points adjacent to a corresponding search point having the smallest SAD if the smallest SAD is one of SAD(56) and SAD(68), finding a smallest SAD among the calculated SADs of the three search points, and performing a third step search for a third step search window using a corresponding search point having the found smallest SAD;

e3) calculating SADs of other two adjacent search points 52 and 54 among four search pints 52, 54, 56, and 58 disposed at top, bottom, left, and right of a search point 5 if the smallest SAD obtained at the step e2) is a SAD of a center search point 5, and finding a smallest SAD among the calculated SADs of the two search points 52 and 54 and the SAD of the center search point 5;

e4) calculating SADs of two search points adjacent to a search point having the smallest SAD if the smallest SAD obtained at the step e3) is one of SAD(52) and

SAD(54), finding a smallest SAD among the calculated SADs of the three search points, and performing a third step search for a third step search window using a corresponding search point having the found smallest SAD; and

e5) performing a third step search for a third step search window using the search point 5 as a center search point if the smallest SAD obtained at the step e4) is the SAD of the center search point 5.

25. The high speed motion estimation method of claim 24, wherein in the step e1), it is determined whether a calculated SAD is larger than a SADmin or not while SADs of search points are calculating, the SAD calculation process stops if the calculated SAD is larger than the SADmin, and a next step is performed.

26. The high speed motion estimation method of claim 24, wherein in the step e1), it is determined whether a calculated SAD is smaller than a predetermined matching error threshold or not, and a corresponding search point of the SAD smaller than the matching error threshold is decided as a motion vector.

27. The high speed motion estimation method of anyone of claims 14 and 15, wherein the second step search includes the steps of:

calculating a SAD of a center search point 3 among nine search points 31 to 39 in the second step search window having the search point 3 as a center, calculating SADs of two adjacent search points 36 and 38 among search points 32, 34, 36, and 38 disposed at a top, bottom, left, and right of the center search point 3, using a three-step search method and finding a smallest SAD among SAD(3), SAD(36), and SAD(38);

calculating SADs of two search points adjacent to a corresponding search point having the smallest SAD if the smallest SAD is one of SAD(36) and SAD(38), finding a smallest SAD among the calculated SADs of the three search points, and performing a third step search for a third step search window using a corresponding search point

having the found smallest SAD;

calculating SADs of other two adjacent search points 32 and 34 among four search pints 32, 34, 36, and 38 disposed at top, bottom, left, and right of a search point 3 if the smallest SAD obtained at the above step is a SAD of a center search point 3, and finding a smallest SAD among the calculated SADs of the two search points 32 and 34 and the SAD of the center search point 3;

calculating SADs of two search points adjacent to a search point having the smallest SAD if the smallest SAD obtained at the above step is one of SAD(32) and SAD(34), finding a smallest SAD among the calculated SADs of the three search points, and performing a third step search for a third step search window using a corresponding search point having the found smallest SAD; and

performing a third step search for a third step search window using the search point 3 as a center search point if the smallest SAD obtained at the above step is the SAD of the center search point 3.

28. The high speed motion estimation method of claim 27, wherein in the steps of calculating the SADs, it is determined whether SAD(3) is smaller than a predetermined SADmin if the smallest SAD is the SAD of a search point 3, and the SADmin is set to SAD(3) if SAD(3) is smaller than the predetermined SAD.

29. The high speed motion estimation method of claim 27, wherein in the steps for calculating the SADs, it is determined whether a calculated SAD is larger than a SADmin or not while SADs of search points are calculating, the SAD calculation process stops if the calculated SAD is larger than the SADmin, and a next step is performed.

30. The high speed motion estimation method of claim 27, wherein in the steps for calculating the SADs, it is determined whether a calculated SAD is smaller

than a predetermined matching error threshold or not, and a corresponding search point of the SAD smaller than the matching error threshold is decided as a motion vector.

31. The high speed motion estimation method of claim 27, wherein the third step search in the steps b22), b24), and b25) includes the step of calculating SADs of all search points in a third step search window using a center search point having the smallest SAD, finding a smallest SAD from the calculated SADs, and deciding a corresponding search point having the smallest SAD as a motion vector.

32. The high speed motion estimation method of claim 24, wherein the third step search in the step e5) includes the step of: calculating SADs of all search points in a third step search window using a center search point having the smallest SAD, finding a smallest SAD from the calculated SADs, and deciding a corresponding search point having the smallest SAD as a motion vector.